

#### REMARKS

Applicants respectfully request favorable reconsideration of this application, as amended.

The specification has been revised to improve the descriptiveness of the title, to address the informalities noted by the Examiner, and to make other minor changes.

Fig. 6 of the drawings has also been revised as suggested by the Examiner. Applicants appreciate the Examiner's careful attention to the application.

By this Amendment, Claims 1 and 13-15 have been cancelled, without prejudice or disclaimer, to reduce the issues. Accordingly, Claims 2-12 remain pending for further consideration, with Claim 2 being independent. Applicants reserve the right to present any of the cancelled claims for further prosecution in a continuing application.

In the outstanding Office Action, independent Claim 2 was rejected under 35 U.S.C. § 102(b) as being anticipated by Damron (U.S. Patent No. 6,108,767). Dependent Claims 3-12 were also rejected over the Damron patent, Claims 3-6 having been rejected for anticipation and Claims 7-15 having been rejected for obviousness.

Without acceding to the rejections on Damron, Claim 2 has been amended more particularly to recite certain distinctive features of Applicants' invention. As now set

forth in Claim 2, a data processor in accordance with Applicants' invention comprises a status register, a central processing unit including a predetermined register set, and a plurality of register banks corresponding to the predetermined register set, the status register including an overflow flag to indicate an overflow of the plurality of register banks. Claim 2 now additionally recites (i) that the central processing unit operates to store information from the predetermined register set to one of the plurality of register banks when an interrupt occurs and the overflow flag indicates non-occurrence of the overflow of the plurality of register banks, and (ii) that the central processing unit operates to store information from the predetermined register set to a stack area when an interrupt occurs and the overflow flag indicates occurrence of the overflow of the plurality of register banks. See, e.g., the specification at page 5, line 2 to page 6, line 4; page 20, first full paragraph; and page 22, first and second full paragraphs.

The Damron reference evidently differs significantly from Applicants' invention as described above. Damron describes a computer architecture that employs a windowed register file (register window file) that contains a set of registers that are available to a program at a particular

moment. Generally, on entering a subroutine, the subroutine invokes a "save" instruction, which provides the subroutine with a new register window. When the register window file overflows, the registers in at least one older register window must be saved (spilled) to memory to release registers for a new register window. Thus, information of the current register set is stored to the new register window, and information from an old register window is moved to another area (for example, a stack area). Damron's teaching of spilling information from older register windows to a memory to free up capacity for a new register window file clearly does not anticipate the above-mentioned features of Applicants' invention. Nor does it render them obvious.

Accordingly, Applicants respectfully request that the outstanding rejections of Claim 2, and its dependents, be withdrawn and that this application now be passed to issue.

The Commissioner is hereby authorized to charge to Deposit Account No. 50-1165 any fees under 37 C.F.R. §§ 1.16 and 1.17 that may be required by this paper and to credit any overpayment to that Account. If any extension of time is required in connection with the filing of this paper and

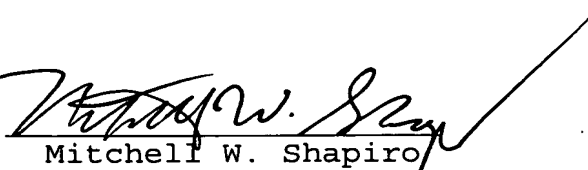
has not been requested separately, such extension is hereby requested.

Respectfully submitted,

MWS:sjk

Miles & Stockbridge P.C.  
1751 Pinnacle Drive  
Suite 500  
McLean, Virginia 22102  
(703) 903-9000

By:

  
Mitchell W. Shapiro  
Reg. No. 31,568

October 16, 2006